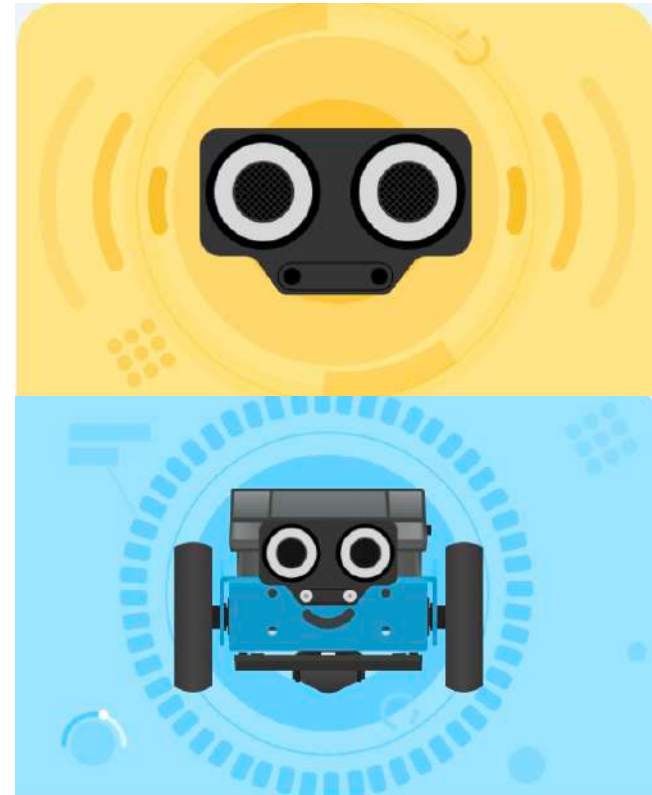




# FICHER 3 DÉTECTION D'OBSTACLES

## Capteur d'ultrasons





## Compétences travaillées dans le cadre du CRCN (Cadre de Référence des Compétences Numériques)

### Compétence 3.4 Programmer

#### De quoi s'agit-il ? (Décret n°2019-919 du 30 août 2019)

Écrire des programmes et des algorithmes pour répondre à un besoin (automatiser une tâche répétitive, accomplir des tâches complexes ou chronophages, résoudre un problème logique...) et pour développer un contenu riche (jeu, site web...) avec des environnements de développement informatique simples, des logiciels de planification de tâches...

#### Références au socle commun de connaissances, de compétences et de culture

- Savoir que des langages informatiques sont utilisés pour programmer des outils numériques et réaliser des traitements automatiques de données. (Domaine 1 – Comprendre, s'exprimer en utilisant les langages mathématiques, scientifiques et informatiques)
- Connaître les principes de base de l'algorithmique et de la conception des programmes informatiques. (Domaine 1 – Comprendre, s'exprimer en utilisant les langages mathématiques, scientifiques et informatiques)
- Mettre en œuvre les principes de base de l'algorithmique pour créer des applications simples. (Domaine 1 – Comprendre, s'exprimer en utilisant les langages mathématiques, scientifiques et informatiques)

#### Thématiques et mots-clés associés (Pix)

Algorithme et programme ; représentation et codage de l'information ; langages de programmation ; complexité ; pensée algorithmique et informatique ; collecte et exploitation de données massives ; intelligence artificielle et robots.



## Compétences travaillées dans le cadre du CRCN (Cadre de Référence des Compétences Numériques)

### Niveaux de maîtrise des compétences numériques et repères pour enseigner

#### Niveau 1

- Lire et construire un algorithme qui comprend des instructions simples

#### Niveau 2

- Réaliser un programme simple

#### Niveau 3

- Développer un programme pour répondre à un problème à partir d'instructions simples d'un langage de programmation
- Modifier un algorithme simple en faisant évoluer ses éléments de programmation
- Mettre au point et exécuter un programme simple commandant un système réel ou un système numérique

#### Niveau 4

- Inscrire l'écriture et le développement des programmes dans un travail collaboratif et constructif
- Modifier le comportement d'un objet régi par un programme simple

#### Niveau 5

- Créer un programme animant un objet graphique ou réel
- Écrire et développer des programmes pour répondre à des problèmes et modéliser des phénomènes physiques, économiques et sociaux



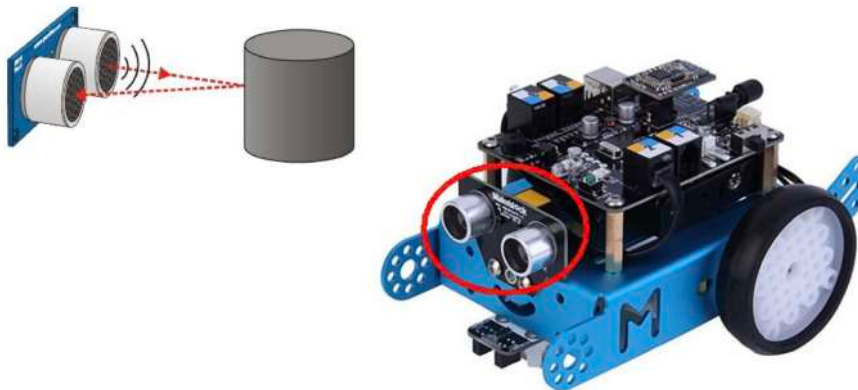
## Installation de l'extension capteur à ultrasons « Ultrasonic Sensor 2 »

Les robots mbot sont munis d'un capteur de distance à ultrason.

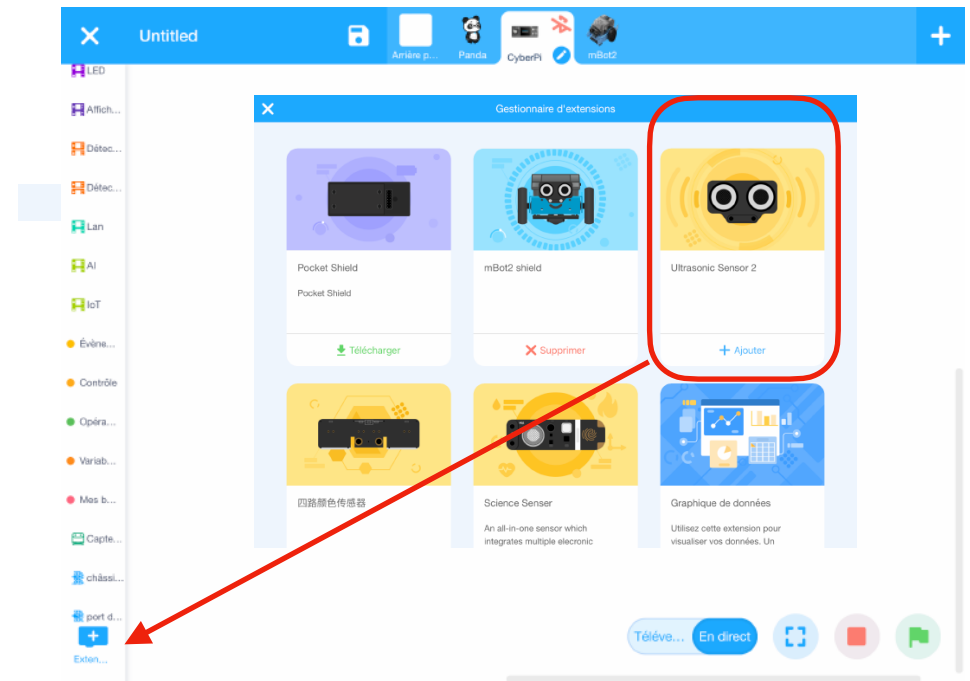
Il fonctionne avec une partie émetteur d'onde et une partie récepteur d'onde.

Cette onde va se déplacer dans l'air et être renvoyé par un obstacle.

La vitesse du son étant de 340 m/s, il est aisé de déterminer la distance de l'obstacle en mesurant le temps écoulé entre le départ de l'ultrason et la réception de l'écho.



Pour installer le module de capteur à ultrasons, cliquer sur **extensions** en bas à gauche de l'interface puis sélectionner «**Ultrasonic Sonic 2**», cliquer sur «**Télécharger**» puis sur «**Installer**».





## Défi obstacles

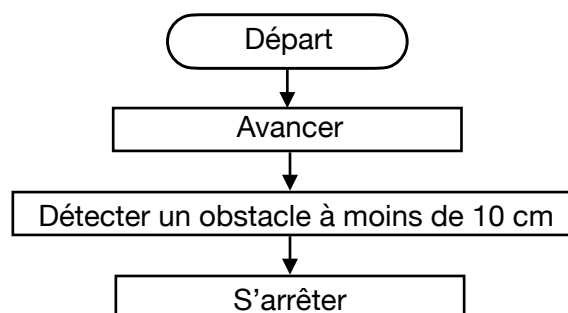
# 1. DÉTECTER UN OBSTACLE ET S'ARRÊTER

**Description :** Le robot avance à une vitesse de 50 tours par minute, s'il détecte un obstacle à moins de 10 cm devant lui, il s'arrête.

Voir le déplacement :



Algorithme :



Briques utilisées :



Programme complet :





## Défi obstacles

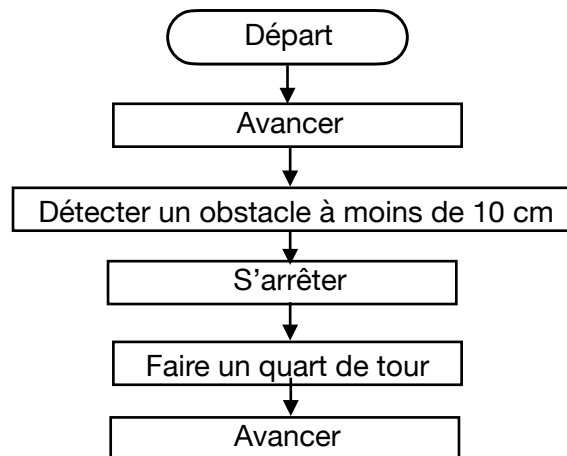
# 2. DÉTECTER UN OBSTACLE, S'ARRÊTER ET FAIRE UN QUART DE TOUR

**Description :** Le robot avance, si la distance détectée est inférieure à 10 cm, le robot s'arrête, effectue un quart de tour, et repart, sinon il avance.

Voir le déplacement :



Algorithme :



Briques utilisées :



Programme complet :





## Défi obstacles

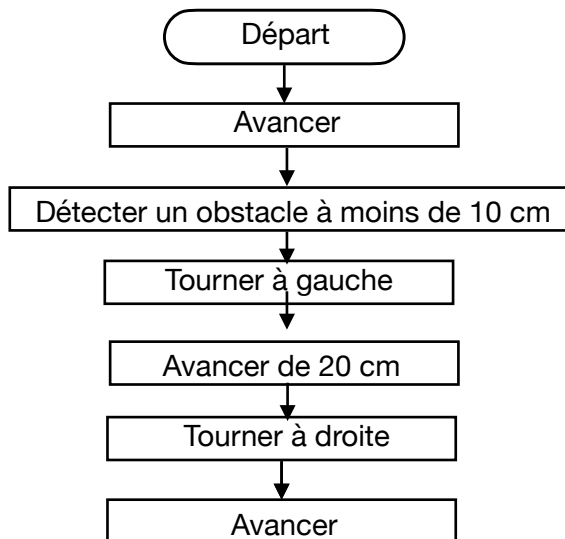
### 3. DÉTECTER UN OBSTACLE ET LE CONTOURNER

Description : Le robot avance, si la distance détectée est inférieure à 10 cm, le robot contourne l'obstacle, puis suit son chemin.

Voir le déplacement :



Algorithme :



Briques utilisées :



Programme complet :





## Défi obstacles

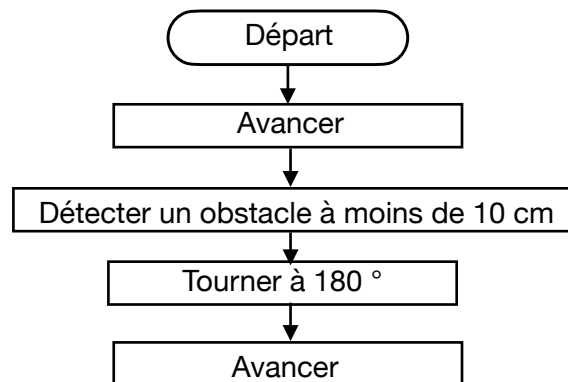
## 4. DÉTECTER UN OBSTACLE ET FAIRE DEMI-TOUR

Description : Le robot avance, si la distance détectée est inférieure à 10 cm, le robot fait demi-tour et avance en sens inverse.

Voir le déplacement :



Algorithme :



Briques utilisées :



Programme complet :





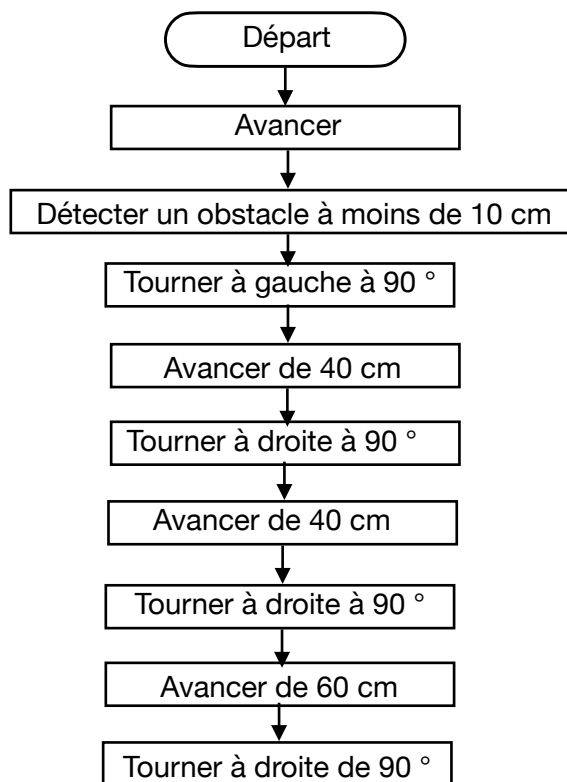


Défi obstacles

## 5. DÉTECTER UN OBSTACLE, FAIRE LE TOUR, REPARTIR EN SENS INVERSE

**Description :** Le robot avance, si la distance détectée est inférieure à 10 cm, le robot fait un tour complet de l'obstacle et repart en sens inverse.

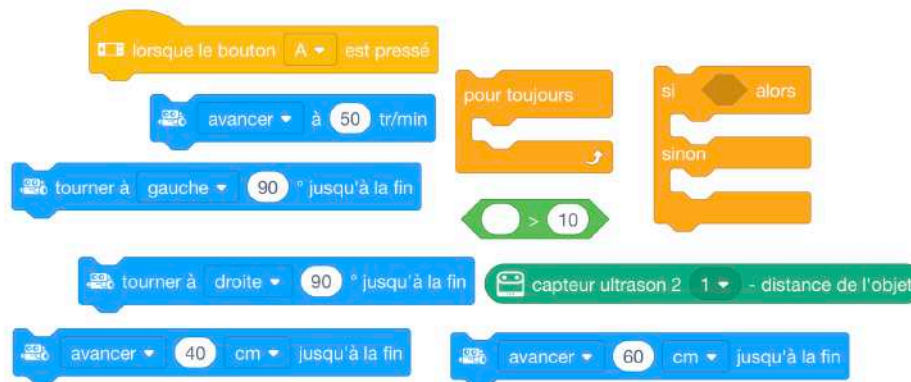
### Algorithme :



Voir le déplacement :



### Briques utilisées :



Certaines briques sont à dupliquer !

### Programme complet :

